WHAT IS CLAIMED IS:

1. A method for decreasing routing latency of a switching platform in a fibre channel network, comprising the steps of:

identifying at least one port from a set of ports in the switching platform, the at least one port having a functional state below a pre-determined threshold state, the pre-determined threshold state being a minimally operational state;

modifying port control instructions associated with the at least one port for reflecting the operational state of the at least one port according to the identification of the at least one port;

operating the switching platform according to the modified port control instructions for reducing polling of the set of ports by polling only ones of the set of ports that indicate being at or above said threshold state.

2. A method implemented in a switch having a plurality of ports, the method comprising:

executing a plurality of instructions for polling a set of operational ports to determine whether each of the operational ports has frames to be routed, wherein the operational ports are polled in a repeating sequential fashion;

examining the plurality of ports to identify one of the operational ports which becomes non-operational;

modifying a set of instructions corresponding to the identified non-operational port so that the identified non-operational port is not polled; and

continuing to execute the plurality of instructions.

3. The method of claim 2, further comprising monitoring the plurality of ports to detect changes in operational states of the plurality of ports.

4. The method of claim 2 wherein modifying the set of instructions corresponding to the identified non-operational port comprises replacing a first instruction in the set of instructions corresponding to the identified non-operational port with a branch instruction.

5. The method of claim 4 wherein the branch instruction causes execution of the plurality of instructions to jump to a set of instructions corresponding to a subsequent port.

6. The method of claim 4 wherein replacing the first instruction comprises writing a data value to the memory address in which the first instruction is stored, wherein the data value comprises a binary representation of the branch instruction.

7. The method of claim 6 wherein the data value is formed by performing a binary OR operation on a branch op code and an offset between the first instruction in the set of instructions corresponding to the identified non-operational port and the first instruction in the set of instructions corresponding to the subsequent port.

8. The method of claim 2, further comprising

examining the plurality of ports to identify a non-operational port which becomes operational;

modifying a set of instructions corresponding to the identified operational port so that the identified operational port is polled; and

continuing to execute the plurality of instructions.

9. The method of claim 8, wherein prior to modification, the set of instructions corresponding to the identified operational port comprise a branch

instruction followed by one or more additional instructions, wherein the branch instruction is configured to cause execution of the plurality of instructions to jump to a set of instructions corresponding to a subsequent port.

10. The method of claim 9 wherein modifying the set of instructions corresponding to the identified operational port comprises replacing the branch instruction with a replacement instruction, wherein the replacement instruction and the one or more additional instructions are configured to poll the identified operational port to determine whether the identified operational port has frames to be routed.

11. The method of claim 9 wherein modifying the set of instructions corresponding to the identified operational port comprises replacing the branch instruction and the one or more additional instructions with a set of replacement instructions configured to poll the identified operational port to determine whether the identified operational port has frames to be routed.

12. The method of claim 10 wherein replacing the branch instruction comprises writing a data value to the memory address in which the branch instruction is stored, wherein the data value comprises a binary representation of the replacement instruction.

13. The method of claim 2, further comprising invalidating at least a portion of an instruction cache after modifying the set of instructions corresponding to the identified non-operational port so that the identified non-operational port is not polled.

14. A method implemented in a switch having a plurality of ports, the method comprising:

executing a plurality of instructions for polling a set of operational ports to determine whether each of the operational ports has frames to be routed, wherein the operational ports are polled in a repeating sequential fashion;

examining the plurality of ports to identify a non-operational port which becomes operational;

modifying a set of instructions corresponding to the identified operational port so that the identified operational port is polled; and

continuing to execute the plurality of instructions.

15. A method for decreasing routing latency of a switching platform, comprising the steps of:

monitoring a set of ports in the switching platform;

identifying ones of the set of ports that undergo a change of operational state; and

modifying port control instructions associated with the identified ones of the set of ports to reflect current operational states of the identified ones of the set of ports;

wherein for each of the set of ports,

if the port is operational, the modified port control instructions associated with the port are configured to poll the port for frames to be routed, and

if the port is non-operational, the modified port control instructions associated with the port are configured to branch to a subsequent port.

16. The method of claim 15 wherein ports having a level of functionality below a predetermined threshold level are considered non-operational and ports having a level of functionality at or above the predetermined threshold level are considered operational.

17. A switch comprising:

a plurality of ports,

wherein each of the plurality of ports has an associated state,

wherein the state of each of the plurality of ports can be either operational or non-operational,

wherein ports in the operational state are configured to receive frames for routing and

wherein ports in the non-operational state do not receive frames for routing;

at least one CPU,

wherein the at least one CPU is configured to execute a polling loop,

wherein the polling loop contains instructions corresponding to each of the plurality of ports,

wherein the instructions in the polling loop corresponding to each of the plurality of ports are configured to poll the corresponding port if the corresponding port is operational and to skip the corresponding port if the corresponding port is non-operational,

wherein the at least one CPU is configured to monitor each of the plurality of ports to identify ones of the plurality of ports that have a change of state and to modify the instructions in the polling

loop corresponding to the identified ones of the plurality of ports to poll the identified ports if the identified ports are operational and to skip the identified ports if the identified ports are non-operational.

18. The switch of claim 17, further comprising an instruction cache configured to store ones of the instructions in the polling loop which were recently executed by the at least one CPU, wherein at least a portion of the instruction cache is invalidated whenever any of the instructions in the polling loop are modified.

19. The switch of claim 17 wherein the at least one CPU is configured to determine that ones of the plurality of ports which have a level of functionality below a predetermined threshold are non-operational and to determine that ones of the plurality of ports which have a level of functionality at or above a predetermined threshold are operational.

20. The switch of claim 17 wherein the at least one CPU is configured to modify the instructions in the polling loop by overwriting one or more instructions in the polling loop with replacement instructions.

21. The switch of claim 20, further comprising a memory coupled to the at least one CPU and configured to store one or more of the replacement instructions.

22. The switch of claim 21 wherein the memory comprises a read-only memory.

23. The switch of 20 wherein upon detection of a port which has become non-operational, the at least one CPU is configured to overwrite a first one of the instructions in the polling loop corresponding to the detected port with a branch instruction, wherein the branch instruction includes a target address of a first one of the instructions in the polling loop corresponding to a subsequent port.

24. The switch of 20 wherein upon detection of a port which has become operational, the at least one CPU is configured to overwrite the instructions in the polling loop corresponding to the detected port with one or more of the replacement instructions, wherein the one or more replacement instructions are configured to poll the detected port and route frames at the detected port.